REMARKS

This Amendment is submitted in response to the Office Action dated June 17, 2004, having a shortened statutory period set to expire September 17, 2004. Claims 1-7 have been amended, claims 8-9 have been cancelled, and claims 10-26 have been added.

Amendments to the Specification

The specification, on page 1, has been amended to include the U.S. Patent Application Numbers of the related and cross-referenced patent applications.

Claim Rejections Under 35 U.S.C. § 101

Claims 1-9 have been rejected under 35 U.S.C. § 101 as being directed to non-statutory subject matter. Applicants respectfully disagree. Data structures embodied in a computer-readable medium are well-accepted as statutory subject matter provided the structure includes a functional relationship. As originally filed, independent claim 1 recited a computer-readable medium having stored thereon a data structure comprising field elements that clearly have a functional relationship. Namely, the first field element was recited as "containing data representing an assigned target" and the second field element was recited as "containing data representing a logic value to be assigned to said assigned target." This relationship (i.e. a target value assignment is amply explained in Applicants' specification with reference to **Figures 12A-12B** (explaining the data structure embodied in instrumentation logic descriptor comments **1223** in assigning a logic value (right hand side expressions) to respective instrumentation signal values (left hand side expressions)). As explained particularly with reference to **Figures 4A-4E** and **12A-12B**, the present invention is directed to a method, system, computer program product and data structure for facilitating the incorporation of instrumentation logic (i.e. diagnostic/testing logic that is used for simulation analysis but is not part of the digital circuit design) in a hardware description language model. The foregoing traversal notwithstanding, amended claims 1-7 and newly added claims 10-26 are clearly directed to statutory subject matter as embodied by the claimed method, system, data structure, and computer program product.

Information Disclosure Statement

Please find attached herewith a completed form 1449 including references to be considered in examining the present application.

Objections to the Drawings

Please find attached herewith a corrected copy of the drawings as requested by the Examiner.

Objections to the Specification

The Specification has been objected to as failing to provide proper antecedent basis for the claimed subject matter, and specifically, that adequate description and characterization for the "intermediate signal" feature recited in claims 6 and 8 is not apparent. As utilized in amended claim 6, the term "intermediate instrumentation signal" is described and characterized in an amended paragraphs beginning on page 90, line 19 and 92, line 15 with reference to signal "Q" depicted in Figures 12A and 12B. Namely, an "intermediate instrumentation signal" is an intermediate (i.e. neither an input nor an output) signal value instantiated within an instrumentation entity (i.e. an entity described in an HDL source code file using a specified syntax that enables HDL compilation means to discern between the instrumentation entity and "design entities" incorporated into the actual design being simulated).

Prior Art Claim Rejections

Claims 1-5, 7 and 9 have been rejected under 35 U.S.C. § 102(e) as being anticipated by U.S. Pat. No. 5,870,309, issued to Lawman (hereinafter *Lawman*). Applicants respectfully traverse the foregoing rejections as they may apply to claims 1-7 as amended herein and newly added claims 10-26 for the following reasons.

The grounds for the foregoing rejections include the observation that *Lawman* discloses HDL circuit descriptions having timing data that is back-annotated in HDL comment lines. *Lawman's* disclosed use of HDL comment lines in this manner provides a useful data repository but does not teach or suggest using such syntax to incorporate functional logic in a simulation model. Applicants' proposed invention is directed to employing a specified syntax (embodied as a modified or "unconventional" HDL comment syntax in the embodiments depicted in **Figures**

3A-3D, **4A-4E** and **12A-12B**) that enables non-design logic to be incorporated into a compiled HDL simulation model while enabling HDL compile and synthesis means to discern between "design entities" that are part of the actual design and "instrumentation entities" (i.e. HDL entities utilized to perform facilitate the testing of the entities incorporated in the digital circuit design) so that, for example, a conventional HDL compiler can be used to compile the model without instrumentation being included in the design. U.S. Pat. No. 6,195,627, commonly owned by the assignee of the present invention, claims an invention wherein a non-conventional comment is utilized to associate an instrumentation entity with a target design entity to be monitored.

As explained in the figures, and particularly with reference to **Figures 4B-4E** and Figures **12A-12B**, Applicants' proposed invention is directed to leveraging the foregoing syntax distinction between design and instrumentation to enable instrumentation logic to be implemented in a design while eliminating the programming and processing overhead required for generating and processing independent HDL files that declare an instrumentation entity. Instead, and as described with reference to **Figures 4B-4E** and **12A-12B**, the present invention uses instrumentation descriptors that employ a specified syntax to achieve the design/instrumentation bifurcation in a flexible and efficient manner.

Amended independent claim 1 now recites in part, a data structure comprising "an assignment statement containing data representing an instrumentation signal and a logic or storage element value assigned to said instrumentation signal;" and "wherein said assignment statement is incorporated in an HDL source code file using a specified syntax that is recognized by an HDL compiler such that the HDL compiler does not instantiate said instrumentation signal into the digital circuit design, and wherein said specified syntax is subsequently processed by an instrumentation load tool to instantiate said instrumentation signal within a design entity described by said HDL source code file." (See assignment statements **1223** in the amended paragraph beginning on page 89, line 12).

Newly added independent claims 10, 15, 18, 21 and 24 incorporate similar limitations. Claim 10 recites, in part, a method comprising a first step of utilizing a "specified syntax to declare instrumentation logic within an HDL source code file," wherein the declaration includes assigning a logic or storage element value to an instrumentation signal; and a second step of

compile processing the HDL source code file, "wherein said compile processing includes processing said specified syntax to determine whether or not to instantiate said instrumentation signal within a design entity described by said HDL source code file."

Independent claim 15 recites a system for implementing instrumentation logic during the model build process as depicted in **Figure 4D**. The system includes "model build processing means for reading an assignment statement containing data representing an instrumentation signal and a logic value assigned to said instrumentation signal, wherein said assignment statement is incorporated in an HDL source code file using a specified syntax;" and "compiler means for compile processing said HDL source code file, wherein said compile processing includes processing said specified syntax to determine whether or not to instantiate said instrumentation signal within a design entity described by said HDL source code file." Independent claim 18 is a program product that mirrors the claims limitations of claim 15.

Independent claim 21 recites a method for implementing instrumentation logic including limitations similar to claim 15. The method includes a first step of "receiving an HDL source code file that describes a design entity, said HDL source code file including an instrumentation signal declaration having an assignment statement that assigns a logic value to an instrumentation signal, wherein said assignment statement is incorporated in the HDL source code file using a specified syntax;" and a second step in which the specified syntax is utilized as a criterion by which compilation means determines whether or not to instantiate (i.e. generate instance data structures depicted in **Figures 4A, 4D and 4E**). Claim 24 recites a system having the substantially the same limitations as claim 21.

Ample support for the foregoing amendment and the event type limitation in amended claim 5 is provided in Applicants' specification on page 30, line 30 through page 31, line 14 (introducing the general concept of "instrumentation entities"), page 32, line 7 though page 33, line 21 (described instrumentation event type categories), page 35, line 12 through page 37, line 27 (describing an HDL source code file **440** that includes instrumentation entity descriptors **451** with reference to **Figure 4C**), page 42, line 1 through page 45, line 27 (with reference to **Figure 4D**, describing a model build process in which bifurcated compilation means embodied by HDL compiler **462** and instrumentation load tool **464** utilizes the instrumentation comment syntax to determine the point in the model build process at which instances of the instrumentation entities

are generated), page 88, line 19 through page 89 (with reference to **Figure 12A**, describing instrumentation logic incorporated within entity **1208** that is processed by compiler means in the manner set forth in **Figures 4B-4E** to determine whether or not to instantiate the logic), page 89, line 11through page 29, line 29 (with reference to **Figures 12A-12B**, describing an HDL source code file **1220** that describes a design entity **1200** and further includes assignment statements in the form of comment lines **1223** that declare instrumentation signals **1240**, **1241**, and **1242**).

By using the aforementioned specified syntax to implement and/or process the instrumentation logic within a given HDL source code description of a design entity, the present invention prevents the instrumentation logic from becoming part of the design, thereby enabling flexible (instrumented or non-instrumented) simulation model processing. Nothing in *Lawman* teaches or suggests any device, article or process whereby comment syntax (or any other specified HDL syntax) is implemented or processed in a manner such that functional diagnostic logic can be instantiated in a simulation model without the instrumentation becoming part of the actual circuit design. Therefore, Applicants submit that amended claim 1, newly added claims 10, 15, 18, 21, and 24, and all claims depending therefrom have been placed in condition for allowance and a notice to that effect is respectfully requested.

No extension of time is believed to be required. However, in the event that an extension of time is required, please charge that extension fee and any other required fees to **IBM Corporation Deposit Account Number 09-0447.**

Applicants invite the Examiner to contact the undersigned attorney of record at (512) 343-6116 if such would further or expedite the prosecution of the present Application.

Respectfully submitted,

Matthew W. Baca
Reg. No. 42,277
DILLON & YUDELL LLP
8911 North Capital of Texas Highway
Suite 2110
Austin, Texas 78759
Telephone (512) 343-6116
Facsimile (512) 343-6446

ATTORNEY FOR APPLICANTS